Notice of Allowability	Application No.	Applicant(s)	
	10/824,042	MACKAY ET AL.	
	Examiner	Art Unit	
	Dennis Myint	2162	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *12/05/2007*.

2. ☒ The allowed claim(s) is/are *1-25 and 30-32*.

3. ☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☒ SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☒ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☒ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☒ Interview Summary (PTO-413), Paper No./Mail Date _____ .
7. ☐ Examiner's Amendment/Comment
8. ☒ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after the Office, on December 6, 2007, granted the application to be withdrawn from issue for consideration of a submission under 37 CFR 1.114 (request for continued examination). As such, Applicant's submission filed on December 5, 2007, has been entered.

2. Claims 1-25 and 30-32 are pending in this office action.

3. In the amendment filed on July 19, 2007, claim 1 was amended. Claims 26-29 had been cancelled. In the amendment filed on December 5, 2007, new drawings are submitted. Claims 1, 11, and 18 are independent claims.

Drawings

1. Drawings of 11 sheets, filed on December 5, 2007, are accepted.

Specification

2. Specification, filed on April 14, 2004, is considered and accepted.

Examiner's Amendment

3. An examiner's amendment to the record appears below. Should the changes and/or additions are be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephonic interview with Attorney George Chaclas on January 3, 2008.

Please amend claims 1, 11, and 18.

1. (Currently Amended) A program product for use in a computer system that executes program steps recorded in a computer-readable medium to perform a method for updating database objects to manage version control of a database configuration in a plurality of database servers in a distributed computing network, the program product comprising:

a recordable media for storing a program; and

the program of computer-readable instructions executable by the computer system to perform steps including:

(a) receiving user specified database schema files for release to a list of corresponding database servers where modifications to occur, wherein each schema file includes proposed database object structures;

(b) verifying that the schema files contain valid Data Definition Language commands;

(c) verifying that the user has proper permissions to modify the database object structures, wherein the proper permissions are controlled by a permission file that specifies which users have permissions to release database object structures and the database servers that respective database object structures are released to;

(d) comparing existing database object structures to the proposed database object structures to determine if the existing database object structures need to be modified;

(e) if the exising database object structures need to be modified, then,

i) automatically generating and executing appropriate commands based on the comparison to modify the existing database object structures;

ii) checking that the database servers being modified are within a respective release time;

iii) automatically generating identification files to identify the database servers that are being managed and an entire set of database object structures being managed, wherein the identification files include release permission information for each database object that specify the users who are able to release changes to each database object, the database servers to which those changes are released and specified times when changes are made to each database server while access to the identification files is limited to administrators;

iv) automatically parsing the valid Data Definition Language commands in a schema file to determine database object structures that a respective database object should have on a database server by database server basis;

v) automatically issuing commands to modify the database object structures of the respective database object on the respective database server so the database object structure matches database object structure that is represented in the respective schema file; and

vi) verifying that the database objects were modified properly such that if any errors occur during a structure change, an original structure of the database object is restored on the respective database server by renaming the original database object by appending a respective object name to an identifier as a backup and renaming the backup back to the original respective object name when an error occurs related to such database object;

(f) automatically creating release notes, based on the identification files, wherein the release notes are generated on a periodic basis and include documentation related to modifications of the database object structures;

(g) automatically releasing permissions files; and

(h) automatically sending the release notes to a plurality of predefined addresses so that users associated with the predefined addresses view a history of where the database objects came from in order to verify reliable storage and, as necessary, retrieval of the database object structures.

11. (Currently Amended) A method for managing a structure of database objects in a plurality of database servers in a distributed computing network such that a configuration of the structure is properly updated so that data therein is reliable, the method comprising the steps of:

(a) receiving user specified database schema files for release to a list of corresponding database servers where modifications to occur, wherein each schema file includes proposed database object structures;

(b) verifying that the schema files contain valid Data Definition Language commands;

(c) verifying that the user has proper permissions to modify the database object structures, wherein the proper permissions are controlled by a permission file that specifies which users have permissions to release database object structures and the database servers that respective database object structures are released to;

(d) comparing existing database object structures to the proposed database object structures to determine if the existing database object structures need to be modified;

(e) if the exising database object structures need to be modified, then,

i) automatically generating and executing appropriate commands based on the comparison to modify the existing database object structures;

ii) checking that the database servers being modified are within a respective release time;

iii) automatically generating identification files to identify the database servers that are being managed and an entire set of database object structures being managed, wherein the identification files include release permission information for each database object that specify the users who are able to release changes to each database object, the database servers to which those changes are released and specified times when changes are made to each database server while access to the identification files is limited to administrators;

iv) automatically parsing the valid Data Definition Language commands in a schema file to determine database object structures that a respective database object should have on a database server by database server basis;

v) automatically issuing commands to modify the database object structures of the respective database object on the respective database server so the database object structure matches database object structure that is represented in the respective schema file; and

vi) verifying that the database objects were modified properly such that if any errors occur during a structure change, an original structure of the database object is restored on the respective database server by renaming the original database object by appending a respective object name to an identifier as a backup and renaming the backup back to the original respective object name when an error occurs related to such database object;

(f) automatically creating release notes, based on the identification files, wherein the release notes are generated on a periodic basis and include documentation related to modifications of the database object structures;

(g) automatically releasing permissions files; and

(h) automatically sending the release notes to a plurality of predefined addresses so that users associated with the predefined addresses view a history of where the database objects came from in order to verify reliable storage and, as necessary, retrieval of the database object structures

18. (Currently Amended) A computer network for managing database comprising:

a plurality of computers, wherein at least one of the computers has a CPU operatively connected to memory for storing databases; and

a source control system on at least one of the computers for managing a structure of database objects in a plurality of database servers in a distributed computing network wherein the source control system is for;

(a) receiving user specified database schema files for release to a list of corresponding database servers where modifications to occur, wherein each schema file includes proposed database object structures;

(b) verifying that the schema files contain valid Data Definition Language commands;

(c) verifying that the user has proper permissions to modify the database object structures, wherein the proper permissions are controlled by a permission file that specifies which users have permissions to release database object structures and the database servers that respective database object structures are released to;

(d) comparing existing database object structures to the proposed database object structures to determine if the existing database object structures need to be modified;

(e) if the exising database object structures need to be modified, then,

i) automatically generating and executing appropriate commands based on the comparison to modify the existing database object structures;

ii) checking that the database servers being modified are within a respective release time;

iii) automatically generating identification files to identify the database servers that are being managed and an entire set of database object structures being managed, wherein the identification files include release permission information for each database object that specify the users who are able to release changes to each database object, the database servers to which those changes are released and specified times when changes are made to each database server while access to the identification files is limited to administrators;

iv) automatically parsing the valid Data Definition Language commands in a schema file to determine database object structures that a respective database object should have on a database server by database server basis;

v) automatically issuing commands to modify the database object structures of the respective database object on the respective database server so the database object structure matches database object structure that is represented in the respective schema file; and

vi) verifying that the database objects were modified properly such that if any errors occur during a structure change, an original structure of the database object is restored on the respective database server by renaming the original database object by appending a respective object name to an identifier as a backup and renaming the backup back to the original respective object name when an error occurs related to such database object;

(f) automatically creating release notes, based on the identification files, wherein the release notes are generated on a periodic basis and include documentation related to modifications of the database object structures;

(g) automatically releasing permissions files; and

(h) automatically sending the release notes to a plurality of predefined addresses so that users associated with the predefined addresses view a history of where the database objects came from in order to verify reliable storage and, as necessary, retrieval of the database object structures.

Allowable Subject Matter

4. Claims 1-25 and 30-32 are allowed. The following is a statement of reasons for the indication of allowable subject matter.

As per claim 1, the prior art of record, alone or in combination, does not teach or fairly suggest the combination steps as recited in the claim. Hotti (hereinafter "Hotti", U.S. Patent Application Publication Number 2002/0169745) in view of Sprenger et al. (hereinafter "Sprenger", U.S. Patent Number 6363388) does not teach the following limitations:

"A program product for use in a computer system that executes program steps recorded in a computer-readable medium to perform a method for updating database objects to manage version control of a database configuration in a plurality of database servers in a distributed computing network, the program product comprising:

a recordable media for storing a program; and

the program of computer-readable instructions executable by the computer system to perform steps including:

(a) receiving user specified database schema files for release to a list of corresponding database servers where modifications to occur, wherein each schema file includes proposed database object structures;

(b) verifying that the schema files contain valid Data Definition Language commands;

(c) verifying that the user has proper permissions to modify the database object structures, wherein the proper permissions are controlled by a permission file that specifies which users have permissions to release database object structures and the database servers that respective database object structures are released to;

(d) comparing existing database object structures to the proposed database object structures to determine if the existing database object structures need to be modified;

(e) if the exising database object structures need to be modified, then,

i) automatically generating and executing appropriate commands based on the comparison to modify the existing database object structures;

ii) checking that the database servers being modified are within a respective release time;

iii) automatically generating identification files to identify the database servers that are being managed and an entire set of database object structures being managed, wherein the identification files include release permission information for each database object that specify the users who are able to release changes to each database object, the database servers to which those changes are released and specified times when changes are made to each database server while access to the identification files is limited to administrators;

iv) automatically parsing the valid Data Definition Language commands in a schema file to determine database object structures that a respective database object should have on a database server by database server basis;

v) automatically issuing commands to modify the database object structures of the respective database object on the respective database server so the database object structure matches database object structure that is represented in the respective schema file; and

vi) verifying that the database objects were modified properly such that if any errors occur during a structure change, an original structure of the database object is restored on the respective database server by renaming the original database object by appending a respective object name to an identifier as a backup and renaming the backup back to the original respective object name when an error occurs related to such database object;

(f) automatically creating release notes, based on the identification files, wherein the release notes are generated on a periodic basis and include documentation related to modifications of the database object structures;

(g) automatically releasing permissions files; and

(h) automatically sending the release notes to a plurality of predefined addresses so that users associated with the predefined addresses view a history of where the database objects came from in order to verify reliable storage and, as necessary, retrieval of the database object structures".

As per claim 11, the prior art of record, alone or in combination, does not teach or fairly suggest the combination steps as recited in the claim. Hotti in view of Sprenger et

al. (hereinafter "Sprenger", U.S. Patent Number 6363388) does not teach the following limitations:

"a method for managing a structure of database objects in a plurality of database servers in a distributed computing network such that a configuration of the structure is properly updated so that data therein is reliable, the method comprising the steps of:

(a) receiving user specified database schema files for release to a list of corresponding database servers where modifications to occur, wherein each schema file includes proposed database object structures;

(b) verifying that the schema files contain valid Data Definition Language commands;

(c) verifying that the user has proper permissions to modify the database object structures, wherein the proper permissions are controlled by a permission file that specifies which users have permissions to release database object structures and the database servers that respective database object structures are released to;

(d) comparing existing database object structures to the proposed database object structures to determine if the existing database object structures need to be modified;

(e) if the exising database object structures need to be modified, then,

i) automatically generating and executing appropriate commands based on the comparison to modify the existing database object structures;

ii) checking that the database servers being modified are within a respective release time;

iii) automatically generating identification files to identify the database servers that are being managed and an entire set of database object structures being managed, wherein the identification files include release permission information for each database object that specify the users who are able to release changes to each database object, the database servers to which those changes are released and specified times when changes are made to each database server while access to the identification files is limited to administrators;

iv) automatically parsing the valid Data Definition Language commands in a schema file to determine database object structures that a respective database object should have on a database server by database server basis;

v) automatically issuing commands to modify the database object structures of the respective database object on the respective database server so the database object structure matches database object structure that is represented in the respective schema file; and

vi) verifying that the database objects were modified properly such that if any errors occur during a structure change, an original structure of the database object is restored on the respective database server by renaming the original database object by appending a respective object name to an identifier as a backup and renaming the backup back to the original respective object name when an error occurs related to such database object;

(f) automatically creating release notes, based on the identification files, wherein the release notes are generated on a periodic basis and include documentation related to modifications of the database object structures;

(g) automatically releasing permissions files; and

(h) automatically sending the release notes to a plurality of predefined addresses so that users associated with the predefined addresses view a history of where the database objects came from in order to verify reliable storage and, as necessary, retrieval of the database object structures".

As per claim 18, the prior art of record, alone or in combination, does not teach or fairly suggest the combination steps as recited in the claim. Hotti in view of Sprenger et al. (hereinafter "Sprenger", U.S. Patent Number 6363388) does not teach the following limitations:

"A computer network for managing database comprising:

a plurality of computers, wherein at least one of the computers has a CPU operatively connected to memory for storing databases; and

a source control system on at least one of the computers for managing a structure of database objects in a plurality of database servers in a distributed computing network wherein the source control system is for;

(a) receiving user specified database schema files for release to a list of corresponding database servers where modifications to occur, wherein each schema file includes proposed database object structures;

(b) verifying that the schema files contain valid Data Definition Language commands;

(c) verifying that the user has proper permissions to modify the database object structures, wherein the proper permissions are controlled by a permission file that specifies which users have permissions to release database object structures and the database servers that respective database object structures are released to;

(d) comparing existing database object structures to the proposed database object structures to determine if the existing database object structures need to be modified;

(e) if the exising database object structures need to be modified, then,

i) automatically generating and executing appropriate commands based on the comparison to modify the existing database object structures;

ii) checking that the database servers being modified are within a respective release time;

iii) automatically generating identification files to identify the database servers that are being managed and an entire set of database object structures being managed, wherein the identification files include release permission information for each database object that specify the users who are able to release changes to each database object, the database servers to which those changes are released and specified times when changes are made to each database server while access to the identification files is limited to administrators;

iv) automatically parsing the valid Data Definition Language commands in a schema file to determine database object structures that a respective database object should have on a database server by database server basis;

v) automatically issuing commands to modify the database object structures of the respective database object on the respective database server so the database object structure matches database object structure that is represented in the respective schema file; and

vi) verifying that the database objects were modified properly such that if any errors occur during a structure change, an original structure of the database object is restored on the respective database server by renaming the original database object by appending a respective object name to an identifier as a backup and renaming the backup back to the original respective object name when an error occurs related to such database object;

(f) automatically creating release notes, based on the identification files, wherein the release notes are generated on a periodic basis and include documentation related to modifications of the database object structures;

(g) automatically releasing permissions files; and

(h) automatically sending the release notes to a plurality of predefined addresses so that users associated with the predefined addresses view a history of where the database objects came from in order to verify reliable storage and, as necessary, retrieval of the database object structures".

The dependent claims, being definite, further limiting, and fully enabled by the section are also allowed.

Contact Information

5. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Dennis Myint whose telephone number is (571) 272-5629. The examiner can normally be reached on 8:30AM-5:30PM Monday-Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Breene can be reached on (571) 272-4107. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Camy Thuong
Primary Examiner

Dennis Myint
Examiner
AU-2162